

## INFORMATION PROCESSOR

**Publication number:** JP2001167058 (A)

**Publication date:** 2001-06-22

**Inventor(s):** KOTANI ATSUSHI

**Applicant(s):** MATSUSHITA ELECTRIC IND CO LTD

**Classification:**

- **international:** G06F15/167; G06F15/16; G06F15/16; (IPC1-7): G06F15/16

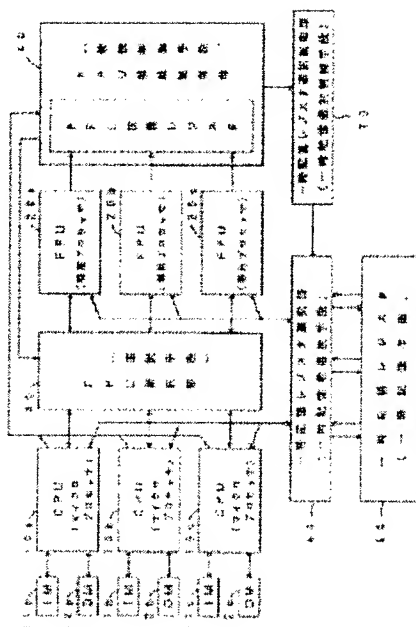
- **European:**

**Application number:** JP19990347779 19991207

**Priority number(s):** JP19990347779 19991207

### Abstract of JP 2001167058 (A)

**PROBLEM TO BE SOLVED:** To shorten the wait time of a microprocessor and to improve throughput concerning an information processor which can perform parallel processing by employing a microprocessor typically exemplified by a CPU and a dedicated processor typically exemplified by a floating-point processing unit (FPU). **SOLUTION:** An information processor has a multi-FPU configuration. The state of FPUs 20a, 20b and 20c are monitored by an FPU state register 42 in an FPU connection control part 40. When a supporting instruction is requested from any one of CPUs 10a, 10b and 10c to an FPU state decoding part 44 in the FPU connection control part 40, an FPU selecting part 30 is controlled so that the FPU in the inactive and idle state can be linked other request CPU on the basis of information in the FPU state register 42. Besides, the trouble of data destruction in the use area of a temporary memory register 50 from a temporary memory register selection control part 70 through the control of a temporary memory register selecting part 60 is prevented.



Data supplied from the esp@cenet database — Worldwide

(19)日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号

特開2001-167058

(P2001-167058A)

(43)公開日 平成13年6月22日(2001.6.22)

(51)Int.Cl.<sup>7</sup>

G 0 6 F 15/16

識別記号

6 2 0

F I

G 0 6 F 15/16

ターミナル\*(参考)

6 2 0 C 5 B 0 4 3

審査請求 未請求 請求項の数7 O L (全 12 頁)

(21)出願番号 特願平11-347779

(22)出願日 平成11年12月7日(1999.12.7)

(71)出願人 000003821

松下電器産業株式会社

大阪府門真市大字門真1006番地

(72)発明者 小谷 敦

大阪府門真市大字門真1006番地 松下電器  
産業株式会社内

(74)代理人 100086737

弁理士 岡田 和秀

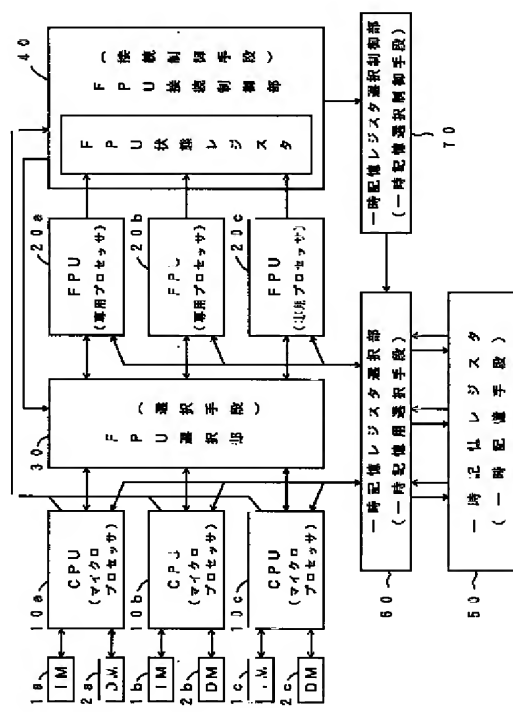
Fターム(参考) 5B045 GC03 GG06 GG08

(54)【発明の名称】 情報処理装置

(57)【要約】

【課題】 CPUを代表例とするマイクロプロセッサとFPU(浮動小数点演算処理装置)を代表例とする専用プロセッサとの並行処理を可能とした情報処理装置において、マイクロプロセッサの待ち時間を短縮し処理能力を向上する。

【解決手段】 マルチFPU構成とする。FPU接続制御部40におけるFPU状態レジスタ42が複数のFPU20a、20b、20cの状態を監視しておく。CPU10a、10b、10cのいずれかからFPU接続制御部40におけるFPU状態解読部44に要支援命令のリクエストがあると、FPU状態レジスタ42の情報に基づいて不動作で空いている状態のFPUをリクエストをしたCPUにつながるようにFPU選択部30を制御する。また、一時記憶レジスタ選択制御部70から一時記憶レジスタ選択部60の制御を介して一時記憶レジスタ50の使用エリアにデータ破壊の不具合が生じないようにする。



## 【特許請求の範囲】

【請求項1】 マイクロプロセッサ以外に複数の専用プロセッサを備え、前記マイクロプロセッサがリクエストを行ったときに前記複数の専用プロセッサのうち不動作状態にある専用プロセッサに対して前記マイクロプロセッサの接続を行うように構成されていることを特徴とする情報処理装置。

【請求項2】 マイクロプロセッサと、複数の専用プロセッサと、前記マイクロプロセッサと前記複数の専用プロセッサのいずれかとの接続状態を選択する選択手段と、前記複数の専用プロセッサの状態を監視しており前記マイクロプロセッサからリクエストがあったときに前記マイクロプロセッサを不動作状態の専用プロセッサに接続するように前記選択手段の制御を行う接続制御手段とを備えていることを特徴とする情報処理装置。

【請求項3】 複数のマイクロプロセッサと、複数の専用プロセッサと、前記複数のマイクロプロセッサのいずれかと前記複数の専用プロセッサのいずれかとの接続状態を選択する選択手段と、前記複数の専用プロセッサの状態を監視しており前記複数のマイクロプロセッサのいずれかからリクエストがあったときにそのリクエストをしたマイクロプロセッサを不動作状態の専用プロセッサに接続する制御を行う接続制御手段とを備えていることを特徴とする情報処理装置。

【請求項4】 前記リクエストをしたマイクロプロセッサからそのデータを受け取って一時記憶し、前記リクエストされた専用プロセッサにそのデータを受け渡す中継の一時記憶手段を備えており、前記リクエストをしたマイクロプロセッサは前記一時記憶手段にデータ転送を完了したのちは独自の処理を実行し、前記マイクロプロセッサから命令を受け取った前記専用プロセッサは前記一時記憶手段をアクセスしながら前記リクエストの命令の処理を実行するように構成されていることを特徴とする請求項3に記載の情報処理装置。

【請求項5】 前記複数のマイクロプロセッサと前記一時記憶手段との間および前記複数の専用プロセッサと前記一時記憶手段との間に介挿されて接続状態を選択する一時記憶用選択手段と、前記接続制御手段からのどの専用プロセッサを接続するかの情報に基づいてその専用プロセッサに対応した領域を前記一時記憶手段に設定しかつその設定領域に対して前記データ転送のために前記リクエストをしたマイクロプロセッサを接続するように前記一時記憶用選択手段を制御する一時記憶選択制御手段とを備えていることを特徴とする請求項4に記載の情報処理装置。

【請求項6】 前記マイクロプロセッサが中央演算処理装置（CPU）であり、前記専用プロセッサが浮動小数点演算処理装置（FPU）などの数値演算プロセッサ（NDP）である請求項1から請求項5までのいずれかに記載の情報処理装置。

【請求項7】 前記複数のマイクロプロセッサとして外部CPUもアクセス可能に構成されていることを特徴とする請求項1から請求項6までのいずれかに記載の情報処理装置。

## 【発明の詳細な説明】

## 【0001】

【発明の属する技術分野】本発明は、中央演算処理装置（CPU）などの主とするマイクロプロセッサ以外に例えば浮動小数点演算処理装置（FPU: Floating-point Processing Unit）などの専用プロセッサ（コプロセッサ）を備え、マイクロプロセッサと専用プロセッサとの並行処理を可能に構成した情報処理装置にかかわり、特に、専用プロセッサの使用効率を高めるとともにマイクロプロセッサの処理能力を高める技術に関するものである。

## 【0002】

【従来の技術】図5は上記のような構成の従来の情報処理装置について、その一例の電氣的構成を示すブロック図である。図5において、符号の511、512はCPU、513、514は命令メモリ（Inst Mem; インストラクションメモリ）、515はデータメモリ（DMEM）、520は一方のCPU512に専属のFPU（浮動小数点演算処理装置）、530は一時記憶レジスタ、600はメインコントロール部（Main Ctrl）、601は命令メモリ（Inst Mem）、602はデータメモリ（DMEM）である。

【0003】この従来技術のマルチCPU方式の情報処理装置は、マイクロプロセッサとしてのCPUが処理すべき負荷が増大したとき、例えば浮動小数点演算を行う必要が生じたときに、その浮動小数点演算を専用プロセッサとしてのFPUにまかせることにし、CPU自身は、その間に別の処理を進める、換言すれば、CPUとFPUとの並行処理が可能な状態に構成されている。

【0004】メインコントロール部（Main Ctrl）600が各CPU511、512に対して、マイクロコードなどの命令群をコンパイル（翻訳）した結果、及び各CPUの動作状態等により、どのCPUにその処理を行わせるかを判断し、装置全体の動作の制御を行っている。例えば、行わせる処理が通常の計算処理つまりFPUを使わなくても完了する処理や、制御系の処理の場合は、各CPU511、512の動作の状態信号より、メインコントロール部600は現在処理の空いている方のCPUへ処理を依頼する。

【0005】行わせる処理が精度を必要とする浮動小数点演算を含む場合は、各CPU511、512の処理の空き具合によらず、付加機能としてFPU521がついている方のCPU512へ処理が依頼される。したがって、FPU521を用いる処理が連続して発生するときには、装置全体が待ち状態になり、先のFPU521を用いる処理が終了した後において初めてCPU512へ

タスクが渡される。

【0006】上記の従来技術の場合には、FPUがつながっているのは特定のCPUであって、他のCPUにはFPUへの使用権がない。

【0007】そこで、個々のCPUに個別に1対1の関係でFPUを接続したマルチCPU・マルチFPU構成の情報処理装置が考えられる。個々のCPUは、他のCPUがFPUを使用しようとしていないかろうと、そのことには無関係に独自にFPUを使用することが可能となる。

【0008】

【発明が解決しようとする課題】上記のうちの前者の特定のCPU512にのみFPU521がつながっている形態の従来技術の場合、次のような問題がある。すなわち、CPU512が浮動小数点演算を含む第1のタスクをFPU521に渡したのち、独自の処理を行い、そのうちに浮動小数点演算を含む第2のタスクが生じたときに、その第2のタスクをFPU521に渡そうとしても、もし第1のタスクが終了していなければ、渡すことができず、第1のタスクが終了するまで待たなければならない。

【0009】例えば処理速度の高速化や扱うデータ量の増大が進んでいるグラフィックス処理を行う情報処理装置においては、特に座標変換等で必要不可欠である浮動小数点演算を含むタスク処理と、他に浮動小数点演算を必要とするタスクが連続的に発生したとき、または次々と座標変換が発生した場合などにおいては、変換しなければならない座標点数に比例して処理の待ち時間が増加し、システム性能の低下を招くといった問題がある。

【0010】また、各CPUごとにFPUが1対1の関係でつながっているようにした従来技術の場合には、FPUを複数備えていても、1つのFPUはそれ専属の1つのCPUにしかつながっておらず、あるCPUにおいてFPUを用いる状況が断続的に発生したときに、そのFPUにおいて先行する浮動小数点演算を含む処理が終了していない限りは、次の浮動小数点演算を含む処理は待ち状態とならざるを得ず、結果的に装置全体の性能低下を招くといった問題がある。

【0011】上記の説明においては、マイクロプロセッサとしてCPUを例にあげ、専用プロセッサとしてFPU（浮動小数点演算処理装置）を例にあげたが、上記のような問題は、CPUとFPUの組み合わせに限らず、広く一般に、マイクロプロセッサと専用プロセッサとの並行処理を可能とした情報処理装置に当てはまることである。

【0012】本発明は上記した課題の解決を図るべく創作したものであって、CPUなどのマイクロプロセッサとFPUなどの専用プロセッサとの並行処理を可能とした情報処理装置において、装置全体の処理能力の向上を図ることを目的としている。特に、浮動小数点演算処

理を行うマルチCPU方式の情報処理装置の処理速度の向上を図ることを目的としている。

【0013】

【課題を解決するための手段】上記した課題の解決を図ろうとする本発明にかかわる情報処理装置は、第一に、専用プロセッサを複数備えることとしている。そして、第二に、複数の専用プロセッサのうち不動作状態つまりは空いている状態にある専用プロセッサを選択的に使用可能となすこととしている。マイクロプロセッサが一つの専用プロセッサを占有している状態において、同じマイクロプロセッサに新たな専用プロセッサの支援を必要とする命令が生じたときは、別の専用プロセッサにその命令の実行を依存することができる。したがって、専用プロセッサに空きがある限りにおいて、換言すれば、空き状態にある専用プロセッサが存在する限りにおいて、マイクロプロセッサは処理を待つ必要がなく、独自の処理の遂行が可能となり、装置全体の処理能力が高まる。

【0014】

【発明の実施の形態】以下、本発明の実施の形態を総括的に説明する。以下において必要に応じて、マイクロプロセッサが専用プロセッサの支援を必要とする命令のことを「要支援命令」と呼ぶことにする。

【0015】本願第1の発明の情報処理装置は、マイクロプロセッサ以外に複数の専用プロセッサを備え、前記マイクロプロセッサが要支援命令についてのリクエストを行ったときに前記複数の専用プロセッサのうち不動作状態にある専用プロセッサに対して前記マイクロプロセッサの接続を行うように構成されている。

【0016】この第1の発明によると、マイクロプロセッサに対して複数の専用プロセッサが用意されている。したがって、マイクロプロセッサは、その複数の専用プロセッサのうち現在不動作となっているすなわち空いている状態の専用プロセッサに対して使用権をもつことができる。そのような使用権を行使できる相手側の専用プロセッサが複数あるので、専用プロセッサの支援を必要とする要支援命令が連続してまたは断続して発生した場合に、一部の専用プロセッサがすでに動作状態に入っている、あるいは課せられた命令の実行が未だ終了していても、他の空いている状態の専用プロセッサを利用することができるため、マイクロプロセッサとしては待機をしなくてもよいことになる。そのような待機しなくてもよい確率が大幅に増大する。このことによって、装置全体の処理能力の向上を図ることが可能となる。

【0017】本願第2の発明の情報処理装置は、マイクロプロセッサと、複数の専用プロセッサと、前記マイクロプロセッサと前記複数の専用プロセッサのいずれかとの接続状態を選択する選択手段と、前記複数の専用プロセッサの状態を監視しており前記マイクロプロセッサからリクエストがあったときに前記マイクロプロセッサを不動作状態の専用プロセッサに接続するように前記選択

手段の制御を行う接続制御手段とを備えた構成となしてある。これは、上記の第1の発明をより詳しく記述したものに相当する。

【0018】この第2の発明による作用は次のとおりである。マイクロプロセッサは専用プロセッサの支援を必要とする要支援命令が発生したときに、接続制御手段に対してリクエストを行う。接続制御手段は複数の専用プロセッサが現在動作状態にあるのか不動作状態にあるのかを監視しているが、マイクロプロセッサからリクエストがあったときの各専用プロセッサの状態を把握して、不動作で空いている状態の専用プロセッサを、リクエストをしたマイクロプロセッサに接続するように選択手段を制御する。

【0019】このような作用により、この第2の発明は、上記の第1の発明について記述したように、専用プロセッサの支援を必要とする要支援命令が連続してまたは断続して発生した場合に、マイクロプロセッサとしては待機をしなくてもよくて、いずれかの空いている状態の専用プロセッサを使うことのできる確率が大幅に増大していることになり、結果として、装置全体の処理能力の向上につながるのである。

【0020】本願第3の発明の情報処理装置は、複数のマイクロプロセッサと、複数の専用プロセッサと、前記複数のマイクロプロセッサのいずれかと前記複数の専用プロセッサのいずれかとの接続状態を選択する選択手段と、前記複数の専用プロセッサの状態を監視しており前記複数のマイクロプロセッサのいずれかからリクエストがあったときにそのリクエストをしたマイクロプロセッサを不動作状態の専用プロセッサに接続する制御を行う接続制御手段とを備えた構成となしたものである。この第3の発明は、上記の第2の発明において、マイクロプロセッサが複数あること、およびそのいずれのマイクロプロセッサも、また選択手段がいずれのマイクロプロセッサに対しても、上記の第2の発明の機能をもつことを記述している。

【0021】この第3の発明による作用は次のとおりである。理解を容易にするために、ここでひとつの例として、マイクロプロセッサが $\mu 1$ と $\mu 2$ の2つあり、専用プロセッサが $\lambda 1$ と $\lambda 2$ の2つあると仮定する。

【0022】この場合、 $\mu 1$ が $\lambda 1$ を占有している状態で、 $\mu 2$ が $\lambda 2$ に使用権を発動することはもちろん可能である。逆に、 $\mu 2$ が $\lambda 1$ を占有している状態で、 $\mu 1$ が $\lambda 2$ に使用権を発動することももちろん可能である。

【0023】 $\mu 1$ が $\lambda 1$ を占有している状態で、 $\mu 1$ に次の要支援命令があったとすると、 $\mu 1$ は $\lambda 2$ に対しても使用権を発動することが可能である。

【0024】同様に、 $\mu 1$ が $\lambda 2$ を占有している状態で、 $\mu 1$ に次の要支援命令があったとすると、 $\mu 1$ は $\lambda 1$ に対しても使用権を発動することが可能である。

【0025】また、 $\mu 2$ が $\lambda 1$ を占有している状態で、

$\mu 2$ に次の要支援命令があったとすると、 $\mu 2$ は $\lambda 2$ に対しても使用権を発動することが可能である。

【0026】同様に、 $\mu 2$ が $\lambda 2$ を占有している状態で、 $\mu 2$ に次の要支援命令があったとすると、 $\mu 2$ は $\lambda 1$ に対しても使用権を発動することが可能である。

【0027】また、別の例として、マイクロプロセッサが $\mu 1$ と $\mu 2$ と $\mu 3$ の3つあり、専用プロセッサが $\lambda 1$ と $\lambda 2$ と $\lambda 3$ の3つあると仮定する。

【0028】この場合、 $\mu 1$ が $\lambda 1$ を占有し、 $\mu 2$ が $\lambda 3$ を占有している状態で、 $\mu 1$ に次の要支援命令があったとすると、 $\mu 1$ は $\lambda 2$ に対しても使用権を発動することが可能である。

【0029】同様に、 $\mu 1$ が $\lambda 1$ を占有し、 $\mu 2$ が $\lambda 3$ を占有している状態で、 $\mu 2$ に次の要支援命令があったとすると、 $\mu 2$ は $\lambda 2$ に対しても使用権を発動することが可能である。

【0030】さらに、別の例として、マイクロプロセッサが $\mu 1$ と $\mu 2$ と $\mu 3$ の3つあり、専用プロセッサが $\lambda 1$ と $\lambda 2$ と $\lambda 3$ と $\lambda 4$ の4つあると仮定する。マイクロプロセッサの個数と専用プロセッサの個数とは必ずしも同じである必要ではなくて、いずれが多くても少なくてもよい。

【0031】この場合、 $\mu 1$ が $\lambda 2$ を占有し、 $\mu 2$ が $\lambda 1$ を占有している状態で、 $\mu 1$ に次の要支援命令があったとすると、 $\mu 1$ は $\lambda 3$ に対しても使用権を発動することが可能であり、さらに $\mu 2$ に次の要支援命令があったとすると、 $\mu 2$ は $\lambda 4$ に対しても使用権を発動することが可能である。

【0032】また、 $\mu 1$ が $\lambda 2$ を占有し、 $\mu 2$ が $\lambda 3$ を占有している状態で、 $\mu 1$ に次の要支援命令があったとすると、 $\mu 1$ は $\lambda 1$ に対しても使用権を発動することが可能であり、 $\mu 1$ にさらに次の要支援命令があったとすると、 $\mu 1$ は $\lambda 4$ に対しても使用権を発動することが可能である。

【0033】以上の例において、複数のマイクロプロセッサと専用プロセッサとが並列処理の状態にあるときに、いずれかの処理が終了して専用プロセッサに空きが生じたとすると、依然として他の専用プロセッサを占有しているマイクロプロセッサにさらなる次の要支援命令があれば、その空きが生じた専用プロセッサに対しても使用権を発動することが可能である。

【0034】このように第3の発明によると、非常にダイナミックで流動的な状態で、個々のマイクロプロセッサは複数の専用プロセッサをきわめて有効に活用することができる。したがって、マイクロプロセッサの待機を極力減らし、要支援命令を直ちに専用プロセッサを用いて実行させることのできる確率が飛躍的に増大し、装置全体の大幅な処理能力の向上を図ることができる。

【0035】本願第4の発明の情報処理装置は、上記の第3の発明において、前記リクエストをしたマイクロプ

ロセッサからそのデータを受け取って一時記憶し、前記リクエストされた専用プロセッサにそのデータを受け渡す中継の一時記憶手段を備えており、前記リクエストをしたマイクロプロセッサは前記一時記憶手段にデータ転送を完了したのちは独自の処理を実行し、前記マイクロプロセッサから命令を受け取った前記専用プロセッサは前記一時記憶手段をアクセスしながら前記リクエストの命令の処理を実行するように構成されているものである。

【0036】この第4の発明によると、マイクロプロセッサは専用プロセッサに対して使用権を発動する際に、その要支援命令の実行に必要なデータを一時記憶手段に転送する。要支援命令のリクエストを受けた専用プロセッサは、その要支援命令の実行に必要なデータをマイクロプロセッサからではなく一時記憶手段から読み込めばよい。その結果として、マイクロプロセッサは専用プロセッサの処理に制約されることなく、その独自の処理を自由に実行することができ、さらなる処理能力の向上を期することができる。

【0037】本願第5の発明の情報処理装置は、上記の第4の発明において、前記複数のマイクロプロセッサと前記一時記憶手段との間および前記複数の専用プロセッサと前記一時記憶手段との間に介挿されて接続状態を選択する一時記憶用選択手段と、前記接続制御手段からのどの専用プロセッサを接続するかの情報に基づいてその専用プロセッサに対応した領域を前記一時記憶手段に設定しかつその設定領域に対して前記データ転送のために前記リクエストをしたマイクロプロセッサを接続するように前記一時記憶用選択手段を制御する一時記憶選択制御手段とを備えた構成となしたものである。

【0038】この第5の発明による作用は次のとおりである。個々の専用プロセッサは一時記憶手段においてそれぞれ特有の領域が割り当てられている。一時記憶選択制御手段はリクエストに応じることとなった専用プロセッサに特有の領域を一時記憶手段において確保するように一時記憶用選択手段を制御する。その結果として、あるマイクロプロセッサとある専用プロセッサとが接続されており、かつ別のマイクロプロセッサと別の専用プロセッサとが接続されている多重の並行処理にあって、一時記憶手段を共用したとしても、各専用プロセッサが使用する領域が明確に区画されていることから、誤ったデータの上書きによって処理に必要なデータが破壊されてしまうといった不具合を確実に防止することができる。

【0039】本願第6の発明の情報処理装置は、上記の第1～第5の発明において、前記マイクロプロセッサが中央演算処理装置（CPU）であり、前記専用プロセッサが浮動小数点演算処理装置（FPU）などの数値演算プロセッサ（NDP：Numerical Data Processor）であるというものである。

【0040】これは、特に、近年の、処理速度の高速化

や扱うデータ量の増大化が進んでいるグラフィックス処理を行う情報処理装置において有効となる。例えば、座標変換等で必要不可欠である浮動小数点演算を含むタスク処理と、他に浮動小数点演算を必要とするタスクが連続的に発生したとき、または次々と座標変換が発生した場合などにおいて、従来技術では変換しなければならない座標点数に比例して処理の待ち時間が増加しシステム性能の低下を招いていたが、この第6の発明によれば、上述の作用のとおり全体処理能力が大幅に向上しているため、そのようなグラフィックス処理にも充分に対応することが可能となる。

【0041】本願第7の発明の情報処理装置は、上記の第1～第6の発明において、前記複数のマイクロプロセッサとして外部CPUもアクセス可能に構成されているというものである。外部CPUに対しても、本装置がもつ資源を有効に活用することが可能となり、トータルの能力アップを図ることができる。

【0042】なお、この明細書において、「CPU」（Central Processing Unit）という記述は、通称の代表的なものを取り上げているだけであって、その他の表現として、例えば「MPU」（Micro Processor Unit）なども含めて、また、専用プロセッサとしては、上記の例以外に、例えば「DSP」（Digital Signal Processor）なども含めて、それぞれ最も広義に解釈し得るものとしてとらえなければならない。

【0043】以下、本発明にかかわるマルチCPU方式の情報処理装置の具体的な実施の形態を図面に基いて詳細に説明する。以下の実施の形態は、マイクロプロセッサとしてCPU（中央演算処理装置）を例にあげ、専用プロセッサとしてFPU（浮動小数点演算処理装置）を例にあげる。

【0044】図1、図2、図3は図示表示として一部ずつが重複した階層的構造をなしている。図1は本発明の実施の形態のマルチCPU方式の情報処理装置における主要構成を抽出して示した基本的な電氣的構成を示すブロック図、図2はマルチCPU方式の情報処理装置の内部におけるより詳しい配線関係を示すとともに、マルチCPU方式の情報処理装置と外部の要素との関係についてより詳しい関係を示すブロック図、図3はマルチCPU方式の情報処理装置の構成要素としてのFPU接続制御部および一時記憶レジスタ選択部についてより詳しい構成を示すブロック図である。

【0045】まず、図1によって本実施の形態のマルチCPU方式の情報処理装置の主要構成について説明する。

【0046】図1において、符号の100は本実施の形態のマルチCPU方式の情報処理装置、200は外部記憶装置、300は外部CPUである。また、10a、10b、10cは互いに独立した状態で動作するマイクロプロセッサの代表例としてのCPU（中央演算処理装

置)、11a, 11b, 11cは各CPU10a, 10b, 10cが実行すべき命令を格納している命令メモリ(インストラクションメモリ:IM)、12a, 12b, 12cは各CPU10a, 10b, 10cが処理に用いるデータを格納するデータメモリ(DM)、20a, 20b, 20cは専用プロセッサの代表例としての浮動小数点演算処理を行うFPU(フローティング演算ユニット)、30は複数のCPU10a, 10b, 10cと複数のFPU20a, 20b, 20cとの制御信号の接続を切り換えるFPU選択部、40はFPU選択部30を制御することによりCPU10a, 10b, 10cとFPU20a, 20b, 20cとの接続状態を制御するFPU接続制御部、42は複数のFPU20a, 20b, 20cのいずれかが動作状態にあるかまたは不動作状態にあるかなどの情報やどのCPU10a, 10b, 10cおよび外部CPU100のうちどのCPUがマスターになっているかの情報を示すFPU状態レジスタである。FPU状態レジスタ42はFPU接続制御部40に含まれているものとして扱ってよい。FPU選択部30が特許請求の範囲にいう「選択手段」相当し、FPU接続制御部40が同じく「接続制御手段」に相当している。

【0047】FPU状態レジスタ42は、FPU20a, 20b, 20cの動作状態を確認し、その動作状態を各CPU10a, 10b, 10cに伝えるように構成されている。

【0048】FPU接続制御部40は、CPU10a, 10b, 10cのいずれかから要支援命令の代表例としての浮動小数点演算処理のリクエストがあったときに、FPU状態レジスタ42の情報を解釈し、不動作となっているすなわち空いている状態のいずれかのFPUとリクエストしているマスターのCPUとを接続するようにFPU選択部30を制御するように構成されている。CPU10a, 10b, 10cとFPU20a, 20b, 20cの接続の関係は、固定的に定まっているわけではなく、常に流動的なものとなる。

【0049】CPUに接続されることとなるFPUは、そのCPUからの浮動小数点演算の命令を受けて、浮動小数点演算を実行するものとして構成されている。

【0050】符号の50はCPU10a, 10b, 10cや外部CPU100のうちマスターとなるCPUからスレイブとなるFPUに対して浮動小数点演算処理に必要なデータを転送する際のデータの一時記憶を行うための一時記憶レジスタである。一時記憶レジスタ50には、FPUに処理を行わせる際の計算対象データや、その他計算に必要な係数等が一時的に格納されるようになっている。この一時記憶レジスタ50が特許請求の範囲にいう「一時記憶手段」に相当している。

【0051】ところで、例えば第1のCPU10aと例えば第3のFPU20cとが接続されている状態で、同

時に、例えば第2のCPU10bと例えば第1のFPU20aとが接続される状態も起こり得る。さらには、第3の接続状態が同時に起こることもあり得る。そのような場合に、あるCPUが一時記憶レジスタ50にアクセスしているときに、別のCPUが同時に一時記憶レジスタ50にアクセスすることになる。そのときに、一時記憶レジスタ50の記憶領域においてデータの上書きでデータの不測の破壊が生じる場合が起こり得る。

【0052】符号の60は、そのような不都合の発生を防止するために、一時記憶レジスタ50とCPU10a, 10b, 10cとの間および一時記憶レジスタ50とFPU20a, 20b, 20cとの間ならびに一時記憶レジスタ50と外部CPU300との間で交通整理的に接続状態を調停(アービトレーション)する一時記憶レジスタ選択部である。また、70はFPU接続制御部40がFPU状態レジスタ42を解釈した結果として一時記憶レジスタ選択部60を上記のように制御する一時記憶レジスタ選択制御部である。一時記憶レジスタ選択部60が特許請求の範囲にいう「一時記憶用選択手段」に相当し、一時記憶レジスタ選択制御部70が同じく「一時記憶選択制御手段」に相当している。

【0053】次に、図2によってマルチCPU方式の情報処理装置100と外部の要素との関係についてより詳しい構成を説明する。

【0054】200はマルチCPU方式の情報処理装置100についての外部記憶装置(DMEM)、300は外部CPU、310は外部CPU300が実行すべき命令を格納している外部命令メモリ(IMEM)、320は外部CPU300の処理に用いるデータを格納する外部データメモリ(DMEM)である。

【0055】81はマルチCPU方式の情報処理装置100におけるCPU10a, 10b, 10cが処理するデータの入出力を行うために外部記憶装置200との通信を制御するデータメモリインタフェース(DMIF)、82はCPU10a, 10b, 10cと外部記憶装置200との間でアクセスの調停(アービトレーション)を行うデータバス調停部(Data Bus Arbiter)である。83はマルチCPU方式の情報処理装置100におけるCPU10a, 10b, 10cと外部CPU300との間で命令やデータのやりとりを行うI/Oインタフェース(I/O IF)である。

【0056】FPU選択部30およびFPU接続制御部40は、マルチCPU方式の情報処理装置100の内部のCPU10a, 10b, 10cに対するのと同様に、外部CPU300とも接続されている。図2において、FPU接続制御部40とCPU10a, 10b, 10cのそれぞれおよび外部CPU300とが往復矢印の接続ライン84, 85で結ばれていることに留意されたい。これは、CPUからのリクエストとFPUからの確認(アクノリッジ)とを示している。



【0057】次に、図3によってFPU接続制御部40および一時記憶レジスタ選択部60についてより詳しい構成を説明する。

【0058】FPU接続制御部40は、FPU状態レジスタ42のほかにFPU状態解読部(DEC; Decoder)44と外部CPUインタフェース46とを備えている。

【0059】CPU10a, 10b, 10cの各々はFPU状態解読部44に対して浮動小数点演算のためのリクエストを送出するようになっている。FPU状態解読部44は、リクエストを受け取ると、FPU状態レジスタ42を参照して、不動作状態となっているFPUがあるかどうか、またそれはどのFPUかを解読し、リクエストを発したCPUに対して、その結果を知らせる。さらに、FPU状態解読部44は、前記のように、FPU20a, 20b, 20cのうち不動作状態のFPUを、リクエストのあったCPUに接続するようにFPU選択部30に対して制御信号を送出する。84aはリクエストの信号ライン、84bはアクノリッジの信号ラインであり、これらは図2の接続ライン84に相当している。

【0060】なお、FPUの不動作状態を判定することにつけて、厳密にFPUの使用状態というのは、FPUが動作している状態を含むのはもちろん、そのときに該当FPUのマスターとなっているCPUが演算用のデータを一時記憶レジスタ50へ転送している状態、および、逆に演算結果を一時記憶レジスタ50から自身のデータメモリへ転送している状態も含むものである。

【0061】一時記憶レジスタ50は、3つのFPU20a, 20b, 20cに対応して3つのバンク50a, 50b, 50cを有している。一時記憶レジスタ選択部60は、CPU10a, 10b, 10cおよびFPU20a, 20b, 20cの各々ならびに外部CPU300からバンク50a, 50b, 50cの各々への書き込みの切り換えを行うための第1の複数のセクタ60a…と、CPU10a, 10b, 10cおよびFPU20a, 20b, 20cの各々からのバンク50a, 50b, 50cの各々の読み出しの切り換えを行うための第2の複数のセクタ60b…と、外部CPU300からのバンク50a, 50b, 50cの各々の読み出しの切り換えを行うための第3のセクタ60cとを備えている。

【0062】一時記憶レジスタ選択制御部70は、一時記憶レジスタ選択部60における各セクタ60a, 60b, 60cを調停的に制御するようになっている。そのことによって、一時記憶レジスタ50において上書きによるデータの破壊を防止している。このことの詳しい動作については後述する。

【0063】次に、上記のように構成された本実施の形態のマルチCPU方式の情報処理装置100の動作を図4のフローチャートに従って説明する。

【0064】ステップS10において外部CPU300が各装置内部のCPU10a, 10b, 10cに行わせるべき処理の割り付けを行う。ステップS20において各CPU10a, 10b, 10cを起動し、処理を開始する。起動をかけられた各CPU10a, 10b, 10cは、ステップS30において浮動小数点演算処理が必要か否かを判断し、浮動小数点演算処理が不要であるときはステップS40に進んで浮動小数点演算を伴わない通常の処理を実行する。そして、ステップS200において終了条件を満たすのを待つ。

【0065】一方、ステップS30において浮動小数点演算処理が必要であると判断したときはステップS10に進んで該当計算命令群をFPU20a, 20b, 20cのいずれかに処理させるべく、当該のCPUはFPU接続制御部40におけるFPU状態解読部44に対してFPU使用のためのリクエストを発行する。

【0066】リクエストを受けたFPU状態解読部44は、ステップS110, S120においてFPU状態レジスタ42の状態を見る。すなわちFPU20a, 20b, 20cのいずれが不動作となっているすなわち空いている状態となっているか否かをモニタする。そして、空いている状態のFPUがないときはステップS130に進んで、リクエストしてきたCPUに対して待機信号を送出する。

【0067】空いている状態のFPUがあったとき、またはFPUに空きが発生したときには、リクエストを許可し、ステップS140において該当CPUから空き状態の使用可能なFPU(20a, 20b, 20cのいずれか)が使用する一時記憶レジスタ50のバンク(50a, 50b, 50cのいずれか)へ演算用のデータを転送する。

【0068】そして、ステップS150において、該当のFPUは、現在のマスターとなっているCPUの命令による浮動小数点演算処理を実行する。このとき、マスターとなっているCPUはFPUを投機的に実行させるものである。したがって、CPU10a, 10b, 10cのうちの任意のCPUと、FPU20a, 20b, 20cのうちの投機的につながっている任意のFPUとの組み合わせには様々なものがある。

【0069】ある1つのCPUとある1つのFPUとがつながって処理を実行している状態で、別のCPUと別のFPUとが並行して処理を実行することが可能であり、さらには、残りのCPUと残りのFPUとが並行して処理を実行することも可能である。

【0070】また、同一のCPUにおいてFPUの支援を必要とする要支援命令が連続してあるいは断続して発生したときに、その同じCPUが異なる複数のFPUに対して同時的あるいは並行的に使用権を発動していることが起こり得る。詳しくは、記述の第3の発明による作用を参照されたい。 $\mu_i$ ,  $\lambda_i$  ( $i=1, 2, \dots$ )で説明



したとおりである。

【0071】個々のFPUにおいて、その浮動小数点演算処理が終了すると、ステップS160においてFPU状態レジスタ42はそれ自身単独のFPUについて処理が終了したことを示す状態に変化し、マスターとなっていたCPUに知らせるが、これに伴って、通知を受けたCPUは演算結果を一時記憶レジスタ50から自身のデータメモリ(12a, 12b, 12cのいずれか)に転送する。そして、ステップS200に進み、終了条件を満たすのを待つ。

【0072】ステップS200において終了条件が満たされたと判断したときは、ステップS210に進んで外部CPU300に対して終了割り込みを行い、次の指令を待つ。すなわち、ステップS220で外部CPU300からの起動を待ち、起動があればステップS20に戻り、また、ステップS230でシステムの終了を待ち、終了がなければステップS220に戻る。

【0073】次に、一時記憶レジスタ50を共用することに起因してのデータ上書きによるデータ破壊の防止について説明する。

【0074】上記したように、一時記憶レジスタ選択部60は誤ったデータの上書きにより、一時記憶レジスタ50に格納されているデータが不測に破壊されないようにするためのものである。

【0075】マルチCPU方式の情報処理装置100におけるCPU10a, 10b, 10cのいずれかがFPU20a, 20b, 20cのいずれかに対して浮動小数点演算系の処理を行わせたい場合、まず演算に必要なデータをCPU自身のデータメモリ(12a, 12b, 12cのいずれか)から一時記憶レジスタ50に転送しなければならない。その際に、他のCPUが使用しているFPUがアクセスしている一時記憶レジスタ50内のアドレス領域に対してデータを上書きしないように、該当CPUはFPU接続制御部40におけるFPU状態解読部44に対してFPUの使用リクエストを出し、それを受けたFPU状態解読部44はFPU状態レジスタ42を参照し、不動作となっているすなわち空いている状態のFPUへの通信を許可する。

【0076】あるFPUに対してあるCPUがマスターとなっており、まだFPU自体には起動をかけていない状態であることをFPU状態レジスタ42が確認した場合、そのことをFPU状態解読部44が解読し、その解読結果に基づいて一時記憶レジスタ選択制御部70を介して一時記憶レジスタ選択部60におけるセクタ60a..., 60b...を制御し、そのFPUがアクセスする領域の一時記憶レジスタ50のバンク(50a, 50b, 50cのいずれか)の入力、出力ポートに対してマスターとなったCPUからの入力、出力バスを接続し、ある決められたアドレス領域のみにアクセスを制限するように制御を行う。

【0077】マスターとなったCPUがFPUに起動をかけた場合は、そのFPUがもともと使用する入力、出力ポートに接続し、FPUにおいて浮動小数点演算処理を行わせる。

【0078】最後にFPUの処理が終了し、結果が一時記憶レジスタ50に残っている状態の場合は、再びマスターとなっているCPUの入力、出力バスが接続され、演算結果を自身のメモリに転送し、転送終了後にFPU接続制御部40に対して起動信号の無効を伝え、以降の処理に入る。

【0079】以上のような接続制御を行うことにより、他のFPUの使用データの破壊を防ぐことが可能となる。

【0080】以上詳しく説明してきたように、本実施の形態の情報処理装置によれば、複数FPUが存在する構成をとり、いずれかのCPUが浮動小数点演算系の命令をすでにあるFPUに処理させている状態であったとしても、その同じCPUに新たな浮動小数点演算系の命令が発生したときには、先のFPUが処理を終えていなくても、別のFPUが空いていればそちらに処理を依頼することが可能となる。したがって、浮動小数点演算処理が多発するようなアプリケーションの場合における従来技術の課題であったところの、あるCPUにのみにしかFPUがついていなかったり、CPUとFPUが1対1でしかつながっていないといったハードウェアの制限で発生する浮動小数点演算系命令の待ち時間の問題をうまく解消することができ、待ち時間の大幅な減少を可能となし、装置全体の性能を向上させることができるに至ったのである。

【0081】以上、本発明の実施の形態について詳述してきたが、本発明は上記の実施の形態に限定される必要性はなく、次のような実施の形態も含み得るものである。

(1) CPUで代表されるマイクロプロセッサとしては、その個数が複数の場合で説明したが、その個数は任意であって、図示の3つの場合以外の任意の複数のほか、単一の場合も含み得るものとする。すなわち、単一のマイクロプロセッサであっても、それに複数の専用プロセッサが選択的に接続される構成の場合を含み得るものとする。

(2) マスターとなったCPUからFPUに要支援命令の実行を依頼するに際して、そのCPUから一時記憶レジスタ50にデータを転送するようにしたが、必ずしもそれにとらわれる必要性はなく、一時記憶レジスタ50を省略して、FPUにそのCPUに付属のデータメモリにアクセスさせるように構成してもよきものとする。

(3) 一時記憶レジスタ50を有するとしても、共通のものではなく、個々のFPUに専用の複数の一時記憶レジスタを設けてもよいし、あるいは、個々のCPUに専用の複数の一時記憶レジスタを設けてもよきものとする。

る。

【0082】上記の(1)～(3)は互いに独立した事項であり、これらのうち任意の事項を任意数適当に組み合わせてもよきものとする。

【0083】本願の明細書または図面に記載してある任意の事項について、その省略の可能性、または特許請求の範囲への追加ならびに発明の詳細な説明の変更の可能性を留保する。

【0084】

【発明の効果】本発明によれば、CPU（中央演算処理装置）が一例とされるマイクロプロセッサに対してFPU（浮動小数点演算処理装置）が一例とされる複数の専用プロセッサを備えておき、専用プロセッサの不動作で空いている状態の確認を経た上での合理的なマイクロプロセッサと専用プロセッサとの結び付きを配分するので、マイクロプロセッサが一つの専用プロセッサを占有している状態において、同じマイクロプロセッサに新たな専用プロセッサの支援を必要とする命令が生じたときは、別の専用プロセッサにその命令の実行を依存することができる。したがって、空き状態にある専用プロセッサが存在する限りにおいて、マイクロプロセッサは処理を待つ必要がなく、独自の処理の遂行が可能となり、装置全体の処理能力を大幅に高めることができる。

【図面の簡単な説明】

【図1】 本発明の実施の形態のマルチCPU方式の情報処理装置における主要構成を抽出して示した基本的な電氣的構成を示すブロック図

【図2】 上記図1の実施の形態のマルチCPU方式の情報処理装置の内部におけるより詳しい配線関係およびマルチCPU方式の情報処理装置と外部の要素との関係についてより詳しい関係を示すブロック図

【図3】 上記図1の実施の形態のマルチCPU方式の情報処理装置の構成要素としてのFPU接続制御部および一時記憶レジスタ選択部についてより詳しい構成を示

すブロック図

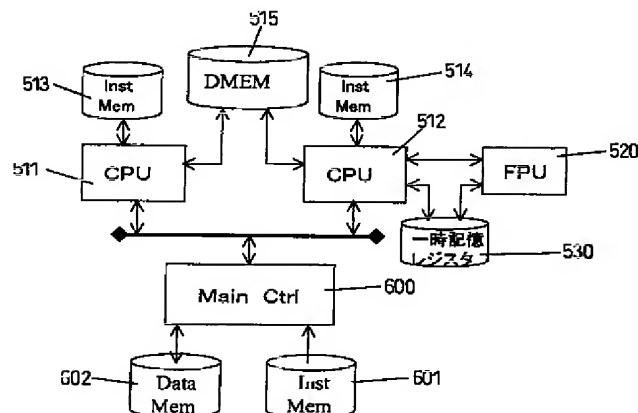
【図4】 上記の実施の形態のマルチCPU方式の情報処理装置の動作を示すフローチャート

【図5】 従来技術のマルチCPU方式の情報処理装置の電氣的構成を示すブロック図

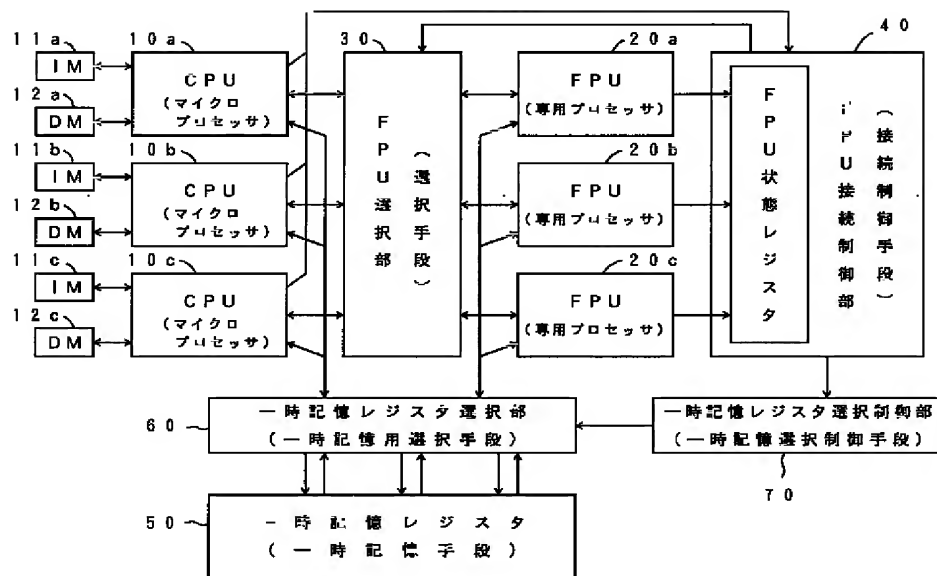
【符号の説明】

10a, 10b, 10c…マイクロプロセッサの一例としてのCPU（中央演算処理装置）  
 11a, 11b, 11c…命令メモリ  
 12a, 12b, 12c…データメモリ  
 20a, 20b, 20c…専用プロセッサの一例としてのFPU（浮動小数点演算処理装置）  
 30…選択手段の一例としてのFPU選択部  
 40…接続制御手段の一例としてのFPU接続制御部  
 42…FPU状態レジスタ  
 44…FPU状態解読部  
 46…外部CPUインタフェース  
 50…一時記憶手段としての一時記憶レジスタ  
 50a, 50b, 50c…バンク  
 60…一時記憶用選択手段としての一時記憶レジスタ選択部  
 60a, 60b, 60c…セクタ  
 70…一時記憶選択制御手段としての一時記憶レジスタ選択制御部  
 81…データメモリインタフェース  
 82…データバス調停部  
 83…I/Oインタフェース  
 84a…リクエストの信号ライン  
 84b…アクノリッジの信号ライン  
 100…マルチCPU方式の情報処理装置  
 200…外部記憶装置  
 300…外部CPU  
 310…外部命令メモリ  
 320…外部データメモリ

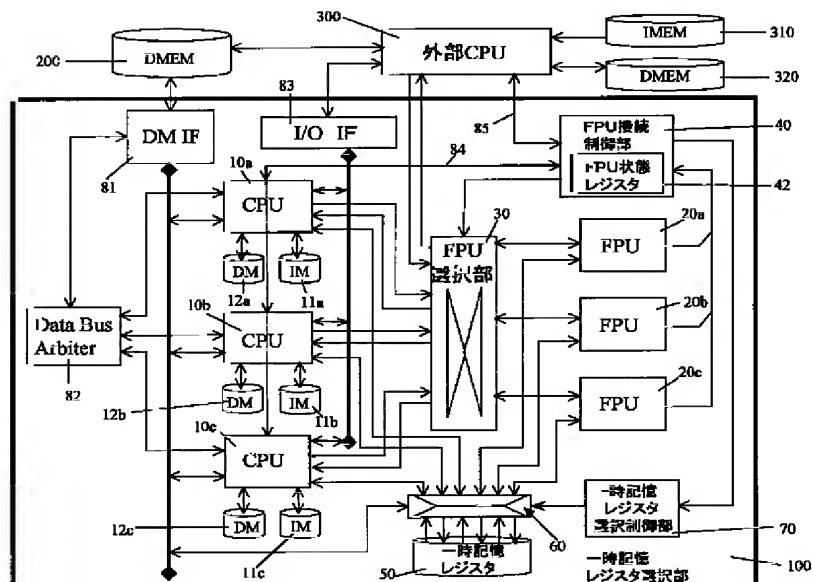
【図5】



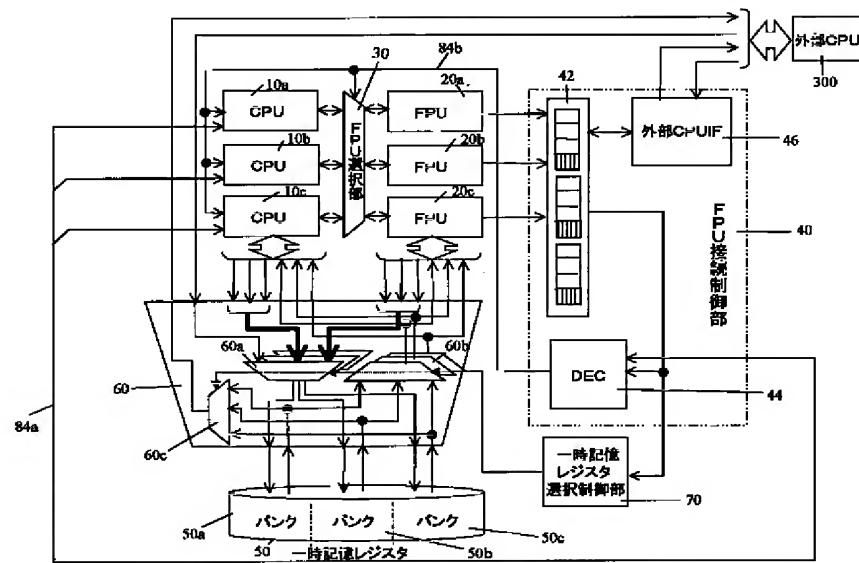
【図1】



【図2】



【図3】



【図4】

